**Design Documentation**

|  |
| --- |
| The American Video Game Company |
| Customer Relationship Management System Proposal |
| Software Solution |
| Tyler Darby  6-16-2022  [Version 0.1] |

Contents

[A. Introduction 3](#_Toc106699387)

[A.1. Purpose Statement 3](#_Toc106699388)

[A.2. Overview of THE PROBLEM 3](#_Toc106699389)

[A.3. Goals and Objectives 3](#_Toc106699390)

[A.4. Prerequisites 3](#_Toc106699391)

[A.5. Scope 4](#_Toc106699392)

[A.6. Environment 4](#_Toc106699393)

[B. Requirements 5](#_Toc106699394)

[B.1. Business Requirements 5](#_Toc106699395)

[B.2. User Requirements 5](#_Toc106699396)

[B.3. Functional Requirements 5](#_Toc106699397)

[B.4. NonFunctional Requirements 5](#_Toc106699398)

[C. SOFTWARE DEVELOPMENT METHODOLOGY 6](#_Toc106699399)

[C.1. Advantages of the waterfall method 6](#_Toc106699400)

[C.2. disAdvantages of the waterfall method 6](#_Toc106699401)

[C.3. Advantages of Agile Method 6](#_Toc106699402)

[C.4. disAdvantages of Agile Method 6](#_Toc106699403)

[C.5. best SUITED 6](#_Toc106699404)

[D. Design 7](#_Toc106699405)

[D.1. Storyboard or Flowchart (Change title to fit needs) 7](#_Toc106699406)

[D.2. UML Diagram (Change title to fit needs) 8](#_Toc106699407)

[D.3. GUI (Change title to fit needs) 9](#_Toc106699408)

[E. Testing 11](#_Toc106699409)

[E.1. Customer editing workflow testing 11](#_Toc106699410)

[E.1.1. Adding a customer 11](#_Toc106699411)

[E.1.2. Update customer button function 12](#_Toc106699412)

[E.1.3. Deleteing a customer 13](#_Toc106699413)

# Introduction

A request has been made by the American Video Game Company to implement a new Customer Relationship Management System. The sections below will outline the needs of the sales force, define clear goals and objectives for this system, give a suggested architecture and suggest a plan for testing.

# A.1. Purpose Statement

This document conveys the professional recommendation for a software solution, a system that meets the needs of a new CRM system for the American Video Game Company’s sales force.

# A.2. Overview of THE PROBLEM

Over the past 2 years the American Video Game Company's sales have been up by 42%, outgrowing its existing CRM systems. To help keep up with increased sales the sales force needs a new system to put into place to keep up. This new system should come in the form of a tool that will help them manage reports, track sales, oversee client contacts and track sales.

# A.3. Goals and Objectives

A new system will be developed in which it carries out existing business processes such as enabling a user to store data, maintain records and their different versions and the many other functionalities the company requires. we will house a server farm and database that will host the system. Any third party users who need access to the system will be able to do so by going to the webpage: AVGCRM.com

# A.4. Prerequisites

Outline any aspects that need to be in place prior to the design, development, and implementation of the project proposed in this document. Be sure to be clear and concise for all listed prerequisites. Also, clearly outline why each prerequisite is needed.

*Note: If no prerequisites are needed, include a paragraph justifying why there are no prerequisites.*

|  |  |  |  |
| --- | --- | --- | --- |
| Number | Prerequisite | Description | Completion Date |
| 0 | n/a | Establish a workflow and needed requirements  Without clear goals and outcomes set the project will not be successful | 6/16/22 |
| 1 | 0 | Requirements listed are approved by the correct body of leadership  without clear and correct requirements, satisfaction from the product will be impossible to achieve | 6/23/22 |
| 2 | 1 | A Budget for the project is developed, approved and obtained | 7/5/22 |

# A.5. Scope

The proposed solution will cover the functional and non-functional company’s and user requirements. A description of the environment, software compatibility and technical specifics as well as its architecture and software specific solution.

# A.6. Environment

The company requires the system be compatible with the following desktop and mobile tablet browsers:

* latest Chrome and Chromium
* latest Firefox
* I.E 9 and above
* Safari 6.0
* mobile & tablet
* iOS7 Safari
* iOS7 Third Party Browsers (Chrome and Firefox)
* Android 4.0 Chrome

The proposed system must integrate with the existing active directory. We will plan to use the currently operating servers in our server farm to host the database and webpage as well as each users specific workstation or tablet / mobile device to access the tool.

# Requirements

The CRM system will meet these 5 requirements listed in the request by the American Video Game Company:

1. Archive information without deletion to maintain historical records

2. Record activity of individual users for auditing and process purposes

3. The system must be user friendly and performance standards must be met

4. A ticketing system that keeps track of every communication and inquiry for contacts

5. produce high level and descriptive data accurate reports

# Business Requirements

Without deleting it from historical records, the CRM system will store all information in a safe archive. Only when a user has privileged permissions, will the CRM system allow them to implement full use of a database that's sole purpose will be for storing information. A clickable button will be on the interface for those who have permission to view this database.

# User Requirements

The system will track individual user activity in a database. There will be a section of the database that will track attempted logins. This as well as other important employee specific information will only be viewable to those with permissions to see such information for the company as it is highly confidential.

# Functional Requirements

A ticketing system will track every communication and inquiry for all contacts including who called, the reason they called the date and time of the call, as well as the follow up to the call and any relevant information. Each instance of a communication and follow up will be uniquely identified. An audit trail will be created for users with the permissions who will be able to access this information.

# NonFunctional Requirements

The system will include large clickable buttons that are clearly visible and labeled for the ease of the user to navigate the system. It will be intuitive while scaling with the amount of users as the system grows over time and is maintained and added to when needed to be.

High level and descriptive reports will be displayed to users with information that is dependent on their permissions to view. Users will have the ability to customize the information of these reports and filter different data depending on data types and categorization information.

# SOFTWARE DEVELOPMENT METHODOLOGY

The company has selected the waterfall software development methodology for this project. Examine the waterfall methodology and compare it to other software development methodologies (e.g., Agile). Include a brief introduction to the development process as well.

# Advantages of the waterfall method

It is well suited for transferring information due to it being so methodical, so it’s easy to transfer information at each step. Due to this it will help different users working on a project have a clear understanding of has been and what still needs to be done. Not only this but it determines the end goal early on in the process and this will help stay committed till the end of the project.

# disAdvantages of the waterfall method

Some disadvantages to the waterfall approach are that the changes are difficult to implement due to the form they take. Testing doesn't start until after completion, this can really be a problem with complex software being developed with this method

# Advantages of Agile Method

The agile method can be great when a group doesn't have a clear plan of action for a project and the design is able to come to fruition through trial and error. Agile can also be a much quicker method than most and can help get products out to the market much quicker.

# disAdvantages of Agile Method

It can be more difficult to measure the progress of a project that is using the agile method than it is a project using the waterfall method due to the lack of a strict schedule. For the same reason projects can be more susceptible to scope creep and can go on for far too long with no end set

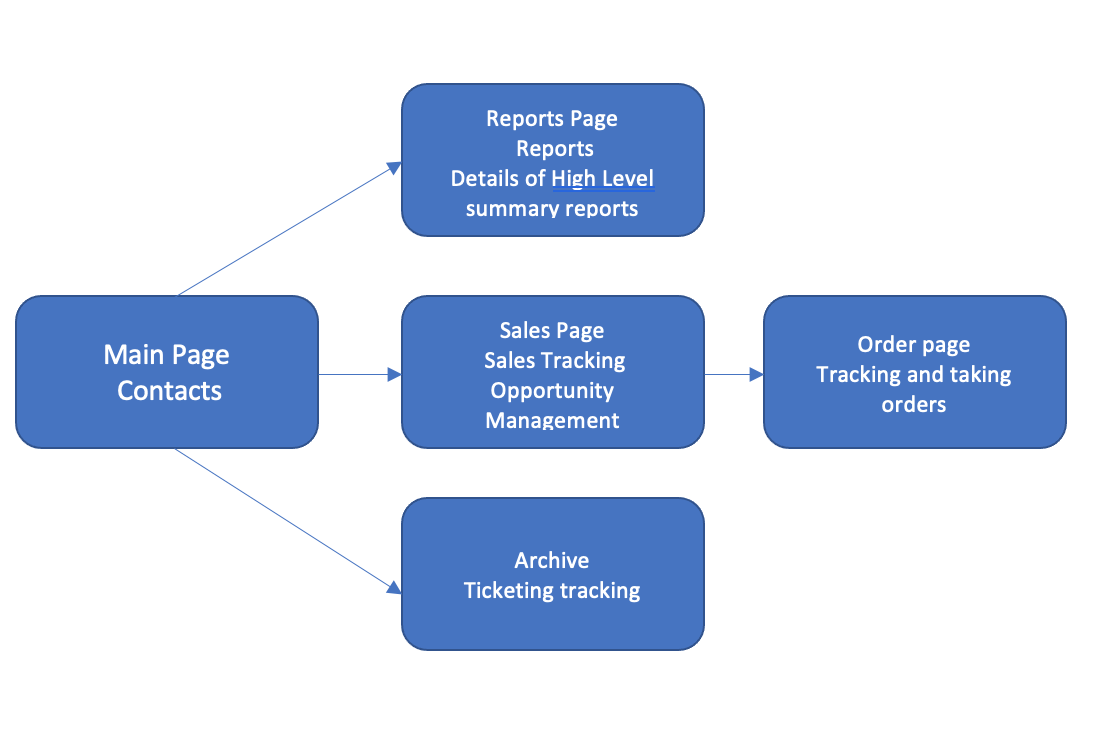
# best SUITED

The waterfall method is best suited for the CRM project for the American Video Game Company instead of the agile method for a few reasons. One it is not in a rush to get to market, if this were the case agile might be more considered but with no competition it is unnecessary to rush the process. Since the project to build the system would be considered predictable there is no need to take advantage of the more flexible design agile offers. Lastly, the project offers clear measurables with a clear outcome for a successful project, due to this its unnecessary to use the agile method over the waterfall method.

# Design

The system will provide a simple and sleek GUI that will be user friendly. The GUI will be easy to navigate and will be modifiable by the different permissions granted to the user that is currently logged into the system.

# Storyboard or Flowchart (Change title to fit needs)



**Figure 1: Sample Storyboard**

# UML Diagram (Change title to fit needs)

This database diagram displays how the primary database will be set up. The system includes a customer class, a staff class and an orders class. The use case diagram shows how the application will be accessible to a user. A user without privileges is able to access contacts, sales and orders while a user with privileges has access to view reports and archive as well as the other pages. The red lines represent users with privileges additional access.

Database Diagram

|  |
| --- |
| **Customer Class** |
| CustomerID |
| firstName |
| lastName |
| phone |
| email |
| ­city |
| state |
| zipCode |

|  |
| --- |
| Staff Class |
| staffID |
| firstName |
| lastName |
| phoneNumb |
| email |
| managerID |

|  |
| --- |
| **Orders class** |
| orderID |
| customerID |
| orderStatus |
| orderDate |
| lastUpdate |
| staffID |

Use Case Diagram

All users access

Privileged users access

Archive

Reports

Orders

Sales

Contacts

# GUI (Change title to fit needs)

Provide a mock-up of the proposed GUI forms that will be used in the proposed solution. Also, clearly indicate where the GUI components point inside the application.

Timeline

Description automatically generated

**Figure 3: Sample GUI Mock-up**

|  |  |  |  |
| --- | --- | --- | --- |
| GUI Control Mapping | | | |
| ID | Control | Property | Data Source |
| 1 | Button | On click open customer page view | Internal Variable |
| 2 | Button | On click open order page view | Internal Variable |
| 3 | Button | On click soft delete highlighted ticket | Internal Variable |
| 4 | Button | On click loads the highlighted customers information into the order page view for updating | Internal Variable, Customer Database, Order Database |
| 5 | TableView | Autofills using customer information | Customer Database |
| 6 | TableView | Autofills using ticket information | Internal Variables, Customer Database and Order Database |

Website

Description automatically generated with low confidence

|  |  |  |  |
| --- | --- | --- | --- |
| GUI Control Mapping | | | |
| ID | Control | Property | Data Source |
| 1 | Button | On click open sales page view | Internal Variable |
| 2 | Button | On click open generated reports page view | Internal Variable |
| 3 | Button | On click soft delete highlighted customer | Internal Variable |
| 4 | Button | On click loads the highlighted customers information into the customer information form for updating | Internal Variable, |
| 5 | Button | On click saves the customers information in the form fields to the database | Internal Variable |
| 6 | Button | On click clears the customers information in the form fields | Internal Variables |
| 7 | TableView | Autofills using customer information | Customer Database |
| 8 | Form | All textfields begin null, can be loaded with text upon click of update customer button | Customer Database, Internal Database |

# Testing

We suggested testing the customer related work flows in the section following

# Customer editing workflow testing

The following work flows will be tested:

Adding a customer  
 Update customer button function   
 Deletion of a customer

# Adding a customer

|  |
| --- |
| Requirement to be tested  Addition of a new Customer |
| Preconditions: Conditions that must be present before test case can successfully run  The database must be created and have the correct schema |
| Steps: The steps the tester must execute to test the feature.   1. Create fake data for a test customer 2. Ensure all fields have values to be tested for their addition 3. Click the *Save Customer Details* Button 4. Check the database and confirm test customer has been added 5. Confirm all information matches related fields |
| Expected results: Expected results and any side effects such as updating a database, writing to a file, etc.  The test customer’s data should match the data created and entered into the form fields during step 1. |
| Pass/Fail: Mark whether the test case passed or failed. The results can be compiled and used to determine if the application is ready for delivery/release.  Pass |

# Update customer button function

|  |
| --- |
| Requirement to be tested  Update Customer button |
| Preconditions: Conditions that must be present before test case can successfully run  Customer data must be loaded into an established database with the correct schema |
| Steps: The steps the tester must execute to test the feature.   1. Select a customer with data in each field in the table 2. Ensure all form fields are blank in form on page 3. Click the “Update Customer” Button 4. Check to ensure data is now loaded into the previously empty form fields 5. Check to ensure data matches the previously selected customer from step 1. |
| Expected results: Expected results and any side effects such as updating a database, writing to a file, etc.  The form on the page should now be filled to match the customers information that was selected for testing in step 1 and should be editable. |
| Pass/Fail: Mark whether the test case passed or failed. The results can be compiled and used to determine if the application is ready for delivery/release.  pass |

# Deleteing a customer

|  |
| --- |
| Requirement to be tested  Deletion of a customer |
| Preconditions: Conditions that must be present before test case can successfully run  Customer data must be loaded into an established database with the correct schema |
| Steps: The steps the tester must execute to test the feature.   1. Select a customer with data in each field in the table 2. Press *Delete Customer* button 3. Check the database and GUI to ensure the customer and respective data is removed |
| Expected results: Expected results and any side effects such as updating a database, writing to a file, etc.  The customer will no longer be expected to be in the database or present in the GUI |
| Pass/Fail: Mark whether the test case passed or failed. The results can be compiled and used to determine if the application is ready for delivery/release.  pass |